

Joni Mökkönen

AUTOSAR

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Ajoneuvotekniikka

Insinöörityö

16.3.2018

Tekijä Otsikko	Joni Mökkönen AUTOSAR
Sivumäärä Aika	31 sivua 16.3.2018
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Ajoneuvotekniikka
Ammatillinen pääaine	Autosähkötekniikka
Ohjaajat	Lehtori Pasi Kovanen
<p>Tavoitteena tässä kirjallisuustutkimuksena suoritetussa opinnäytetyössä oli tutkia AUTOSAR-arkkitehtuuria, jotta selviää, mikä AUTOSAR AUTomotive Open System ARchitecture on ja mitä se sisältää. Työssä esitellään standardoitua arkkitehtuuria kehittävää AUTOSAR-organisaatiota ja sen historiaa. Myös väylätekniikkaan perehdytään sen historian ja kehityksen muodossa. AUTOSARin standardeja ja vaiheita esitellään pääpiirteissään. Arkkitehtuuria käsitellään kokonaisuuden lisäksi järjestelmä- ja sovellusohjelmien tasoilla.</p> <p>Viestintää ja tiedonsiirtoa käsitellään niihin liittyvien uudehkojen ja ajankohtaisten tekniikoiden osalta. Näitä ovat muun muassa virran- ja kustannussäästöllisesti merkittävät tekniikat: partial- ja pretended networking ja domain-pohjaiset verkkorakenteet, joiden tuntemus esimerkiksi ajoneuvojen vianhaussa on tarpeen.</p> <p>Aiheen tutkimisen, tiedonhaun ja raportoinnin lisäksi kerätyn aineiston käytännöllisyyttä opetuksessa arvioidaan, sekä pohditaan tutkimuskohteesta saadun tiedon vastaavuutta tavoiteltuun. Työn lopputuloksena muodostui suomenkielinen katsaus AUTOSARIin ja siihen liittyviin tekniikoihin. Työhön kerättyjä materiaaleja on mahdollista hyödyntää aiheen opiskelussa.</p>	
Avainsanat	AUTOSAR, partial ja pretended networking, domain-pohjaiset verkkorakenteet, väylätekniikka

Author Title	Joni Mökkönen AUTOSAR
Number of Pages Date	31 pages 16 March 2018
Degree	Bachelor of Engineering
Degree Programme	Automotive Engineering
Professional Major	Automotive Electronics Engineering
Instructors	Pasi Kovanen, Senior Lecturer
<p>The purpose of this Bachelor's thesis, which was carried out as a literature research, was to study AUTOSAR architecture, and to find out what AUTOSAR AUTomotive Open Sys-tem ARchitecture is, and what it contains. In this thesis the organization and the history of the AUTOSAR organization, which develops standardized architecture, is introduced. This thesis also discusses the development and history of bus technology. The main points of standards and the phases of the AUTOSAR are presented. The whole architecture is explained more specifically by system and application software level.</p> <p>Communication is introduced with regard to new and current technologies. These energy and cost efficient technologies are among other things partial and pretended networking and domain based networks which are important to know for example when working with diagnostics.</p> <p>In addition to studying the subject, information retrieval and reporting practicality of collected material for educational purposes are analyzed. Also equivalency between goals and results of the thesis are discussed. As a result of the thesis a Finnish review of AUTOSAR and thereto related technologies were created. It is possible to use the materials collected for this thesis when studying AUTOSAR architecture for educational purposes.</p>	
Keywords	AUTOSAR, partial ja pretended networking, domain-based networking, bus technology



Sisällys

Lyhenteet

1	Johdanto	1
2	AUTOSAR	2
2.1	AUTOSARin perustaminen	2
2.2	AUTOSAR organisaationa	3
3	Standardit ja arkkitehtuuri	5
3.1	Standardit yleisesti	5
3.2	Arkkitehtuuri	6
3.3	Sovellusohjelmien arkkitehtuuri	8
3.4	Järjestelmäohjelmisto arkkitehtuuri	9
4	AUTOSARin viimeisimmät julkaisut ja vaiheet	13
4.1	Vaihe 1	13
4.2	Vaihe 2	13
4.3	Vaihe 3	16
5	Väylätekniikka	18
5.1	Analogisesta tiedonsiirrosta digitaaliseen	18
5.2	Eri väylät vaatimuksineen	19
5.3	LIN	20
6	Viestintä ja tiedonsiirto	21
6.1	Partial ja pretended networking	21
6.2	Ethernet	22
6.3	Domain-pohjaiset verkkorakenteet	22
6.4	Viestintäpalvelut	24
6.5	Tietoturvallisuus	26
7	Kerätyn aineiston analysointi	28
8	Yhteenveto	29
	Lähteet	30



Lyhenteet

ECU	Electronic Control Unit. Elektroninen ohjainyksikkö, joka on komponentti osana ajoneuvon tietoväylää.
TTP	Time Triggered Protocol
CAN	Controller Area Network
LIN	Local Interconnect Network
SAE	Society for Automotive Engineers
MOST	Media Oriented Systems Transport
AUTOSAR	AUTomotive Open System ARchitecture
I-PDU	Interaction Protocol Data Unit
N-PDU	Network Protocol Data Unit
RTE	Run time Environment
COM	Component Object Model
API	Application Programming Interface



1 Johdanto

Tämä insinöörityö on tehty Metropolia Ammattikorkeakoulun opetuksen mahdollista kehitystarvetta silmällä pitäen. Idea ajankohtaisen ja tuoreen aiheen tutkimisen tarpeelle löytyi ajoneuvojen datan ja diagnostiikan kanssa toimivan yrityksen henkilöstöltä tiedustelemalla.

Työn tarkoituksena oli selvittää, millainen ajoneuvojen sähköjärjestelmien standardoitu arkkitehtuuri AUTOSAR AUTomotive Open System ARchitecture on, esitellä sen kehittämisen historiaa, tutkia domain-pohjaisia verkkorakenteita, partial ja pretended networkingiä sekä tietoturvallisuutta ajoneuvoissa. Tutkimus- ja selvitystyön tavoitteena on lisäksi pohtia, onko ajoneuvosähkötekniikan opetuksessa Metropolia Ammattikorkeakoulussa tarvetta käsitellä aihetta nykyistä tarkemmin ja laajemmin.

Nykytilanne opetuksessa on se, että AUTOSAR olemassaolo mainitaan kurssilla ajoneuvosähkötekniikka 1, mutta asiaa ei käsitellä sen enempää (Kovanen Pasi 2017).

Kirjallisuustutkimuksena suoritettussa työssä esitellään lyhyesti ajoneuvosähkötekniikan historiaa ja tekijöitä, jotka johtivat AUTOSARin perustamiseen. AUTOSAR-organisaation esittelyn lisäksi työssä esitellään lyhyesti AUTOSAR-standardin kehittymistä kokonaisuutena.

Työssä tutustutaan AUTOSAR-arkkitehtuurikonseptiin ja sen kerroksiin yleisellä tasolla. Aiheeseen perehdytään syvällisemmin viestinnän ja tiedonsiirron osalta. Kyseinen raja on tehty, koska osa-alue on havaittu voimakkaasti kehittyväksi ja täten ajankoh-
taiseksi.

Työn lopussa tulkitaan, pohditaan ja analysoidaan työssä saatuja tuloksia. Työssä kerätyn aineiston käyttömahdollisuudet koulutuksessa analysoidaan tarkemmin ja pohditaan, minkä verran tietoa aiheesta ajoneuvosinöörit tarvitsevat työelämässä erityisesti Suomen työmarkkinoilla.

2 AUTOSAR

2.1 AUTOSARin perustaminen

Ajoneuvojen sähköisten järjestelmien arkkitehtuurin monimutkaistumisen jatkuva kasvusuuntaus johtuu innovatiivisten ajoneuvosovelluksien yleistymisestä. Näitä ovat esimerkiksi: kuljettajaa tukevat järjestelmät, informaatiojärjestelmät ja verkkoon kytketyt palvelut, joita kuluttajat uusilta ajoneuvoilta vaativat. Samaan aikaan ajoneuvojen kehitysvaatimuksiin sisältyy polttoaineen kulutuksen pienentäminen ja lainsäädännöllisten ympäristöstandardeiden mukaisten ajoneuvojen kehittäminen. (Learning Module AUTOSAR 2016:1–2.)

AUTOSAR (AUTomotive Open System ARchitecture) eli ajoneuvojen avoin arkkitehtuurijärjestelmä perustettiin ajoneuvovalmistajien ja niiden alihankkijoiden toimesta vuonna 2003. AUTOSARin taustalla on tavoite kehittää arkkitehtuuri ohjainlaitteiden ohjelmistoille. Tällä pyritään vastaamaan nykyaikaisissa ajoneuvoissa tarvittavien entisestään monimutkaistuvampien ohjelmistojen kehitystarpeeseen. (Learning Module AUTOSAR 2016:1–2.)

AUTOSARin standardit mahdollistavat, sen että niiden mukaisesti valmistetut ECU ohjelmistoinen ovat vähintäänkin samankaltaisia ja yhteensopivia keskenään. Tällä päästään eroon yrityskohtaisista ratkaisuksista, joita muutoin jouduttaisiin käyttämään, kuten ennen standardeja (Bosch Automotive Electrics and Automotive Electronics 2007: 160). Myös perustoiminnot järjestelmissä ovat samankaltaisia ja yhteensopivia. Tästä huolimatta kilpailevien uusien ominaisuuksien kehittämiseksi ja toteuttamiselle on jätetty tilaa. (Learning Module AUTOSAR 2016:1–2.)

Standardeilla määritellään ajoneuvoihin kehitettävälle ohjelmille perustavanlaisia runkoja, jotka ovat saatavilla helposti organisaatioon kuuluville yrityksille. Tämä mahdollistaa ohjelmiston jakamisen ECU-verkkoon joustavasti, jolloin koko järjestelmä tuotantokustannuksineen on optimoitavissa. (Learning Module AUTOSAR 2016:1–2.)

2.2 AUTOSAR organisaationa

AUTOSAR-standardin kehittäjiin kuuluu ajoneuvovalmistajia, ajoneuvovalmistajien ensisijaisia tavarantoimittajia ja muita yrityksiä ohjelmisto-, puolijohde- ja elektroniikkateollisuudesta. AUTOSAR-kehitysorganisaatioon kuuluvat tahot on jaettu viiteen eri ryhmään. Jäsenyyksiä ovat ydin-, premium-, kehitys- ja osakaskumppanuus sekä osanottajat. (Current Partners 2017.)

Ydinpartnereihin lukeutuvat AUTOSARin perustaneet yhdeksän suurta yritystä, jotka ovat BMW Group, Bosch, Continental, Daimler, Ford, GM, PSA, Toyota ja Volkswagen. Premium-partnereita on yhteensä 50 kappaletta. Premium-partnereihin kuuluu pienempiä ajoneuvovalmistajia, merkittäviä alihankkijoita ja muita isoja alan toimijoita. Premium-partnereilta vaaditaan joko vuotuinen jäsenmaksu tai riittävästi työpanosta. (Current Partners 2017.)

Kehityskumppanuuteen vaaditaan vuotuinen jäsenmaksu ja työpanosta. Kehityskumppaneita ovat pienemmät alan yritykset ja start-upit. Kehityskumppaneita vuoden 2017 loppupuolella on 32 kappaletta. (Current Partners 2017.)

Osakaskumppanuuteen vaaditaan vuosimaksu ja sitä vastaan saa valmiit standardit käyttöön. Osakaskumppaneita on 111 kappaletta, joiden joukossa on muun muassa pienempiä ajoneuvovalmistajia, alihankkijoita ja muita alan yrityksiä eri puolilta maailmaa. Kyseiset yritykset eivät osallistu kehitysyhteistyöhön, mutta valmistavat ja/tai käyttävät standardin mukaisia tuotteita. (Current Partners 2017.)

Osanottajia AUTOSARIin on yhteensä 16 kappaletta ja niihin lukeutuu muun muassa muutama yliopisto ja tuotekehitysyrityksiä. Osanottajilta ei peritä vuosimaksuja, mutta tapauskohtaisesti vaaditaan yhteistyötä työryhmissä kehitys- ja premiumpartnereiden kesken. (Current Partners 2017.)

AUTOSAR tarjoaa ajoneuvovalmistajille muun muassa mahdollisuuden jakaa kehitystä tavarantoimittajien kesken, yksinkertaistaa ohjelmien ja järjestelmien yhteensopimista, kilpailla uusilla toiminnoilla suunnittelun ollessa joustavaa sekä ylipäänsä minimoida tuotekehityskustannukset. (Current Partners 2017.)

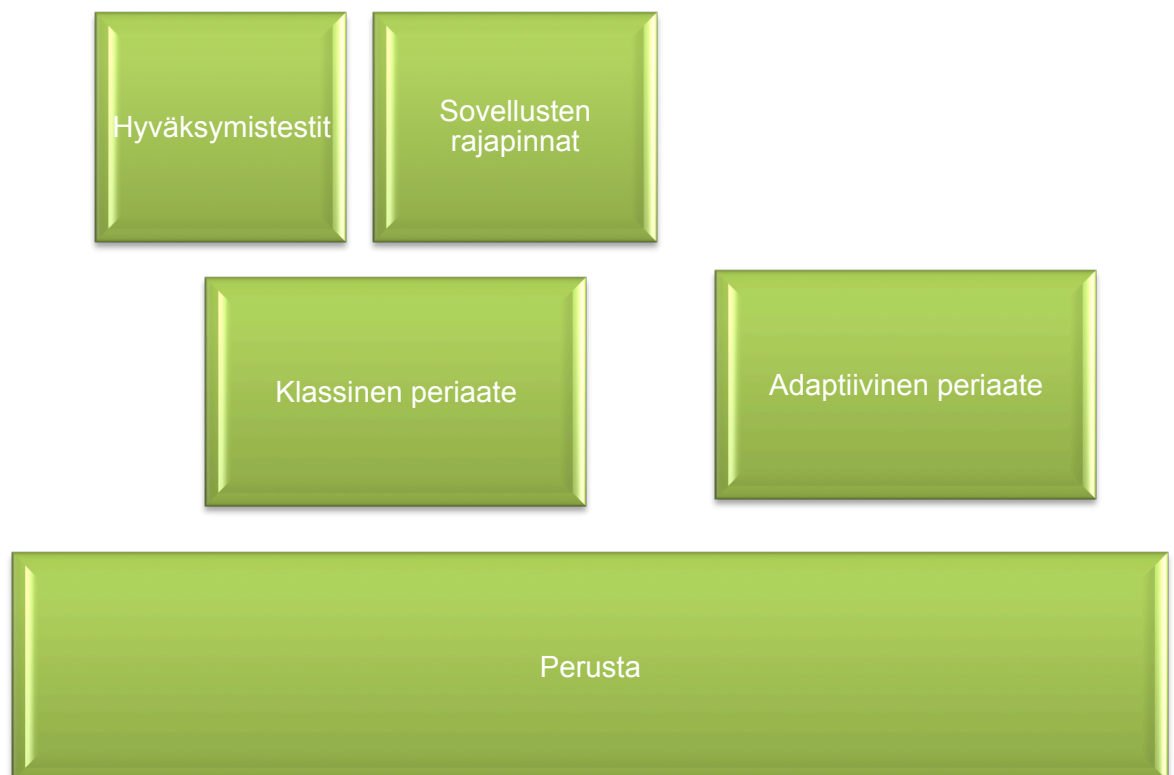
Tavarantoimittajille AUTOSAR tuo etuina esimerkiksi mahdollisuuden myydä samoja tuotteita eri ajoneuvovalmistajille, minimoida erilaiset versiot tuotteista ja tilaisuuden keskittyä kehittämään erilaisia uusia sovelluksia. (Current Partners 2017.)

Työkalujen tuottajille AUTOSAR mahdollistaa selkeän rajapinnan kehitysprosesseissa. Markkinoille pyrkivät yritykset voivat AUTOSARin ansiosta ottaa käyttöön uusia liiketoiminnan malleja standardoitujen rajapintojen puitteissa. (Current Partners 2017.)

3 Standardit ja arkkitehtuuri

3.1 Standardit yleisesti

AUTOSARin standardit on jaettu viiteen eri kategoriaan, joita on jaettu edelleen pienemmiksi osiksi. Kategorioita ovat klassinen periaate, adaptiivinen periaate, hyväksymistestit klassista periaatetta varten, sovelluksien rajapinnat ja perusta. Kuvassa 1 on esitetty standardien kategoriat kerroksittain. (Standards 2017.)



Kuva 1. Standardien jaottelu (Standards 2017).

Klassista periaatetta noudatetaan sulautetuissa järjestelmissä, joissa on tiukat rajoitteet reaaliaikaisuudessa ja turvallisuudessa. Adaptiivista periaatetta hyödynnetään puolestaan suuren suorituskyvyn omaavissa ohjainlaitteissa, jotta saadaan rakennetuiksi järjestelmiä esimerkiksi autonomista ajamista varten. (Standards 2017.)

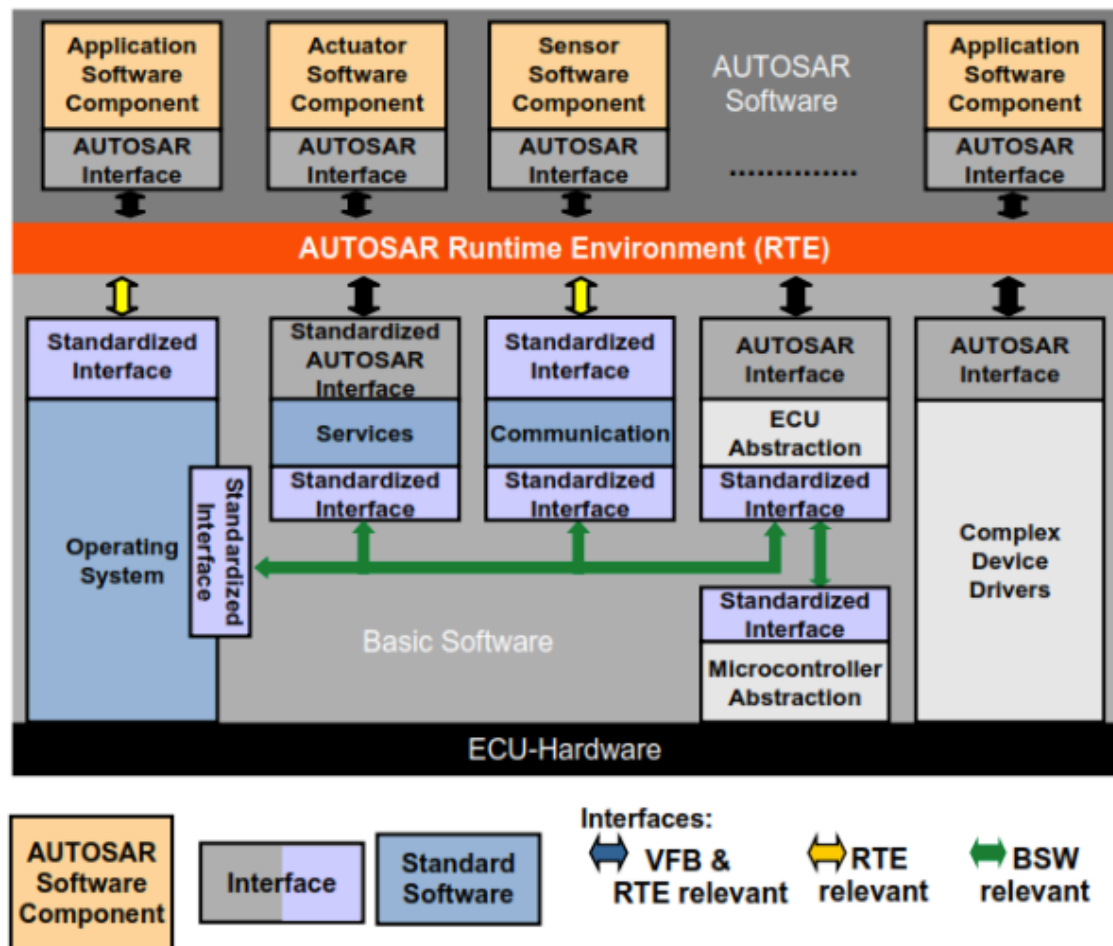
AUTOSARin perustan standardit pitävät sisällään yleisiä osia klassisesta ja adaptatiivisesta periaatteesta. Näitä yleisiä osia ovat muun muassa yleiset näkökohdat metodologiaan ja väyläprotokollat. (Standards 2017.)

Klassisen periaatteen hyväksymistestien standardoidut järjestelmätestit ovat testejä, joita suoritetaan väylä ja ohjelmatasoilla. Tarkoituksena on vahvistaa testattavien komponenttien käyttäytyminen osana tietoväylää ja ohjelmistokokonaisuutta. (Standards 2017.)

AUTOSARin standardisoimat lukuisat ohjelmien rajapinnat on eritelty kuulumaan viiteen eri ajoneuvon osa-alueeseen: kori- ja turvallisuus, moottorin voimalinja, voimansiirron voimalinja, alustan hallinta ja matkustajien sekä jalankulkijoiden turvallisuus. (Standards 2017.)

3.2 Arkkitehtuuri

AUTOSAR määrittelee ohjelmistoarkkitehtuurin sulautettuna ECUun ja se tarjoaa toimintamallin joka on jaettu kolmeen perusosaan: sovelluskerros, perusohjelmisto ja ajonaikainen ympäristö, joka tarjoaa standardisoidun ohjelmistorajapinnan sovelluksen ohjelmistoon. Kuvassa 2 on esitetty arkkitehtuurikerrokset ja niiden suhteet toisiinsa. Ajonaikaisen ympäristön (RTE) yläpuolella kuvassa on sovelluskerros ja sen alapuolella laitteiston päällä on perusohjelmisto. (Zurawski 2014: 1289.)



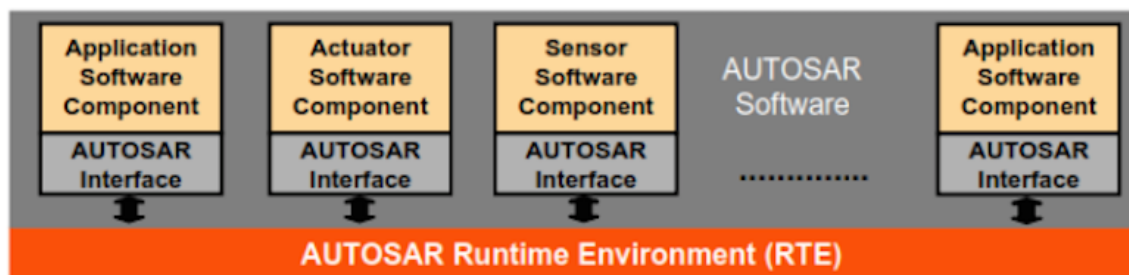
Kuva 2. AUTOSAR:in arkkitehtuurikerrokset (Maradana 2015: 1).

Sulautettujen järjestelmien laadun ja luotettavuuden parantamiseksi käytetään hyvin sovitettua erittelyä sekä esimerkkimallin tukia ohjelmiston ja laitteiston välillä. Nämä helpottavat hallitsemaan järjestelmien monimutkaistumista, sekä mahdollistavat sovel-lusohjelmien komponenttien käytettävyyden eri kohteissa. Toimivalla erittelyllä mahdol-listetaan lisäksi joustavuus tuotteen muokattavuudessa, päivittämisessä ja parantelus-sa, kuten myös sen tuottamistavoissa. (Zurawski 2014: 1289.)

AUTOSAR ohjelmistoarkkitehtuurissa erotellaan jyrkästi sovellusohjelmat, perusohjel-mat ja järjestelmäohjelmat. Perus- ja järjestelmäohjelmat tarjoavat perustan jolle sovel-lusohjelmat rakennetaan. Perusohjelmilla toteutetaan toiminnallisuutta kuten viestintä-protokollat ja diagnostiikka moduulit. Ajonaikainen ympäristö käyttäytyy puolestaan sovellusohjelmien ja järjestelmän ohjelma moduulien välisenä rajapintana. Se toimii myös infrastruktuuripalveluina, jotka mahdollistavat sovellusohjelmakomponenttien välisen viestinnän. (Zurawski 2014: 1309.)

3.3 Sovellusohjelmien arkkitehtuuri

Kuvassa 3 esitetty ylimpänä harmaalla pohjalla oleva sovelluskerros pitää sisällään sovellusohjelmistokomponentit rajapintoihin. AUTOSARin mukaiset sovellusohjelmat ovat rakennettu sovellusohjelmakomponentteihin sisälle. Ne kommunikoivat porttien välityksellä, jotka ovat yhdistetty liittimin. AUTOSARin mukaiset sovellusohjelmat koostuvat uusiokäytettävistä ohjelmistokomponenteista. Kokoonpanollisten tekijöiden vuoksi AUTOSAR helpottaa sovellusohjelman hierarkkista hajauttamista erottelemalla ydinohjelmistokomponentit ja muokattavammat rakenneohjelmistokomponentit. (Zurawski 2014: 1309.)



Kuva 3. Sovelluskerros ja ajonaikainen ympäristö. (Maradana 2015: 1).

AUTOSAR määrittelee erityyppiset ydinohjelmistokomponentit, joita ovat:

- sovellusohjelmistokomponentit
- parametriohtelmistokomponentit
- palveluohjelmistokomponentit
- palvelu proxy ohjelmistokomponentit
- monimutkaiset ajuri ohjelmistokomponentit
- ohjainlaitteenkuvailu ohjelmistokomponentit
- anturitoimilaite ohjelmistokomponentit

- pysyvän lohkon ohjelmistokomponentit.

Parametri- ja sovellusohjelmakomponentit ovat uudelleensijoitettavissa eri ohjainlaitteisiin. Muun tyyppiset ohjelmistokomponentit puolestaan sidottuja tiettyihin ECUihin, koska ne tarvitsevat erityisen osan laitteistoa toimiakseen esimerkiksi jonkin anturin tai muistikomponentin. (Zurawski 2014: 1310.)

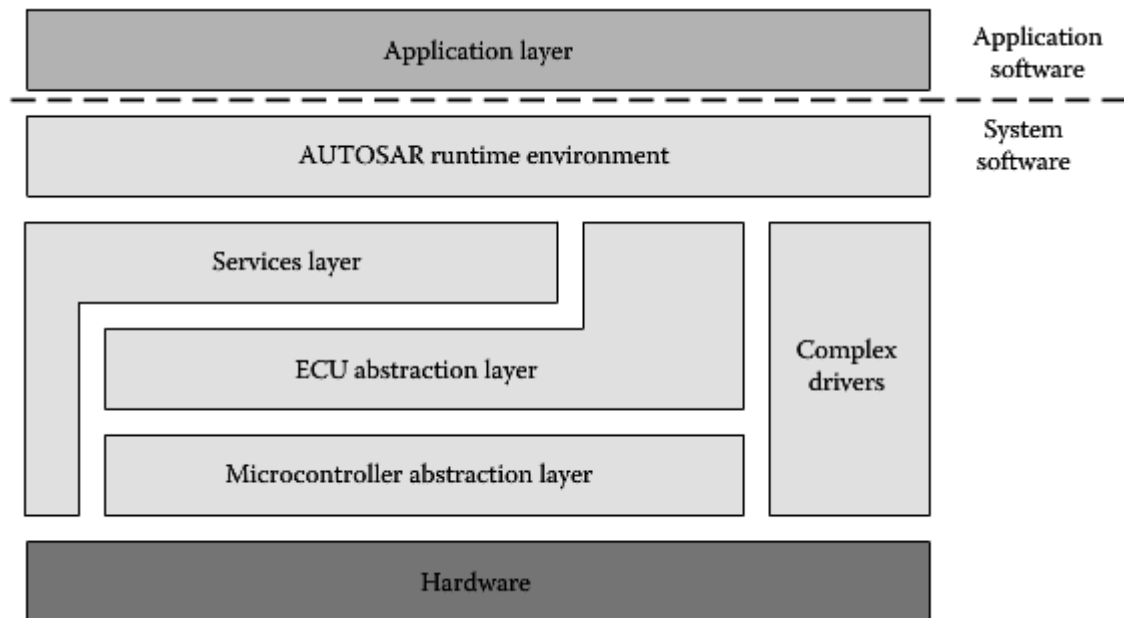
Kaikkien ohjelmistokomponenttien käyttäessä portteja ympäristöihinsä viestiessään, riippuu porttien tyyppi siitä, vaativatko vai tarjoavatko ne palveluita vai erityisiä tietoelementtejä. AUTOSAR määrittelee ja jaottelee portit niiden liitântätapojen tyyppien perusteella. Lähettäjä-vastaanottajaliitännässä yksittäinen lähettäjä jakaa tietoa yhdelle tai useammalle vastaanottajalle tai yksi vastaanottaja vastaanottaja saa tietoa vähintään yhdeltä lähettäjältä. Kyseisessä liitântätavassa muodostetaan tietoelementtejä, jotka määrittelevät vaihdettavaa tietoa, joka voi olla yksinkertaista kuten arvoja tai monimutkaista kuten lukujoukkoja. (Zurawski 2014: 1310.)

Asiakas-palvelinliitântätavassa vähintään yksi asiakas voi herättää palvelimen toimintaan. Tämänkaltaisen viestintä voi olla synkronoitua tai asynkronoitua. Tilankytkentä liitântätavassa puolestaan tilan hallitsija julkaisee vallitsevan tilan käyttäjille, jotta tilankäyttäjät säätävät toimintaansa tilan vaatimalla tavalla. Tilankytkentä liitântätapa muodostaa tilanilmoitusryhmän, johon kerätään kaikki ilmoitukset. (Zurawski 2014: 1310–1311.)

Parametriliitântätavan avulla ohjelmistokomponentti julkaisee tietoa kalibroinneista tai muuttumattomista asioista. Parametri- ja pysyvän lohkon liitännöissä muodostetaan tietoelementtejä samaan tyyliin kuin lähettäjä-vastaanottajaliitännässäkin. Laukaisinliitännässä ohjelmistokomponentti kykenee suoraan laukaisemaan täytöntöönpantavia juoksevia kokonaisuuksia muilta ohjelmistokomponenteilta, jotta tietyt tapahtumat saadaa suoritettua nopeasti. (Zurawski 2014: 1311.)

3.4 Järjestelmäohjelmisto arkkitehtuuri

AUTOSAR määrittelee järjestelmäohjelmistoille kerrostetun arkkitehtuurin, joka tarjoaa perustan sovellusohjelmistokomponenttien toimeenpanolle. Kuvassa 4 on esitetty järjestelmäohjelmistomodulien kerrokset karkeasti jaoteltuina kerroksina.



Kuva 4. AUTOSAR–arkkitehtuurikerrokset (Zurawski 2014: 1313).

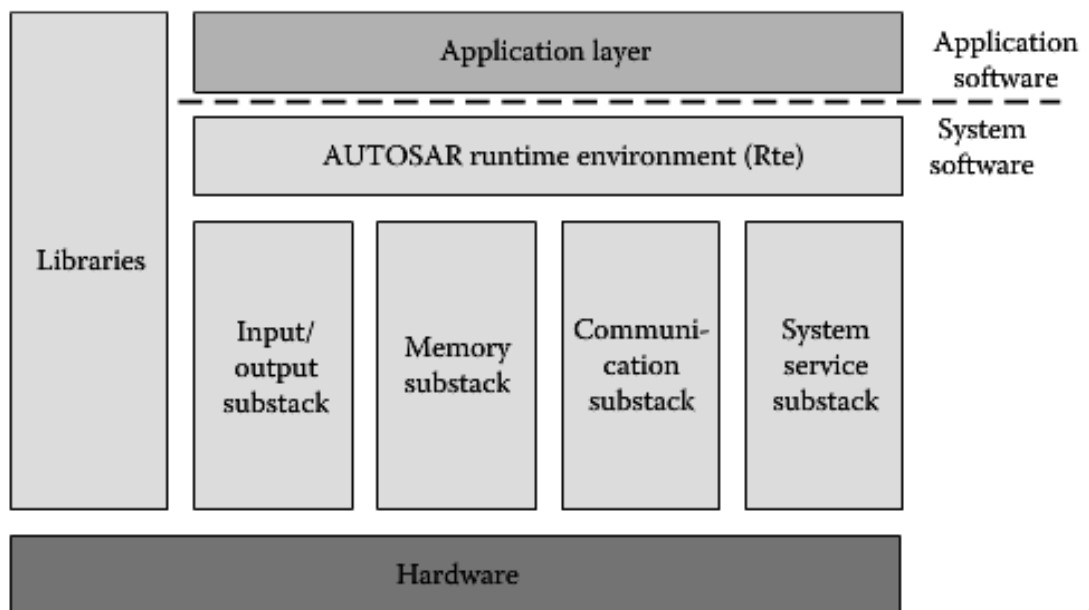
Kuvassa alimpana laitteiston päällä oleva mikrokontrollerikerros sisältää ohjelmistomoduuleja, jotka ovat suoraan yhteydessä sisäisiin oheislaitteisiin ja muistikartoitettuihin ulkoisiin laitteisiin. Kerroksen avulla ylemmät ohjelmistokerrokset itsenäistetään mikrokontrollerin tyyppistä riippumattomiksi. (Zurawski 2014: 1312.)

ECUkerros puolestaan liittää ohjelmistomoduulit mikrokontrollerikerrokseen ja mahdollisesti myös sisällyttää ulkoisten laitteiden ajureita. Kerros tarjoaa standardisoidun sovelluksen ohjelmoinnin rajapinnan, jolla tavoittaa ulkoisia laitteita liitäntöjä käyttäen. Enimmäkseen huipulle sijoitettu palvelukerros tarjoaa käyttöjärjestelmäpalveluita, ajoneuvon verkkoviestinnän ja hallinnan palveluita, muisti palveluita, diagnostiikkaa sekä tilanhallintapalveluita koko ohjainlaitteelle. Kerroksen tehtävä on mahdollistaa peruspalvelut muille järjestelmäohjelmistomoduuleille ja sovellusohjelmistolle. (Zurawski 2014: 1312–1313.)

Yhdistelmäajuri kerros erottuu hajottamalla arkkitehtuurin kerrostuneisuutta tarjoamalla mahdollisuuden hyväksikäyttää jo olemassa olevia laiteajureita AUTOSAR järjestelmäohjelmistomoduuli osiossa. Lisäksi konsepti helpottaa sovellus- ja ohjainlaiteriippuvaisien ajureiden liitettävyyttä. Nämä vaativat usein antureiden tulkintaa sekä toimilaitteenhallintaa ja pääsyä mikrokontrolleriin muun muassa suorituskyvylisistä syistä, jolloin kerrostettu arkkitehtuuri olisi haitaksi. (Zurawski 2014: 1313.)

AUTOSARin ajonaikainen ympäristö tarjoaa liitännällisyyden sovellusohjelmakomponenttien ja järjestelmäohjelmistomoduulien välille. Se mahdollistaa myös infrastruktuuripalveluita, joiden avustamina viestintä sovellusohjelmakomponenttien välillä mahdollistuu. AUTOSARin kirjastoja ei ole liitetty erikseen mihinkään yksittäiseen ohjelmakerrokseen, vaan kaikilla kerroksilla on pääsy niihin. (Zurawski 2014: 1313.)

Kuvan 5 mukaisesti järjestelmäohjelmistot voidaan esittää myös eri alaryhmissä, joita ovat järjestelmäpalvelu-, viestintä-, muisti-, sisäänmeno- ja ulostuloalaryhmä. Järjestelmäpalveluihin sisältyy ohjainlaitteille räätälöidyt järjestelmäpalvelut ja kirjastotoiminnot. Viestintäalaryhmään kuuluu ohjelmistomoduuleja, jotka tarjoavat standardinmukaisen pääsyn väylille. (Zurawski 2014: 1313.)



Kuva 5. Järjestelmäohjelmistojen alaryhmät (Zurawski 2014: 1314).

Muistialaryhmä käsittelee ohjelmistomoduuleja, jotka toimivat sisäisten ja ulkoisten muistien kanssa. Sisäänmenoryhmä ja ulostuloalaryhmä keskittyvät antureihin ja toimilaitteisiin liittyvien ohjelmistomoduulien standardisointuun liitääntään. (Zurawski 2014: 1313.)

AUTOSARin järjestelmäohjelmistomoduuleihin on myös sovellettavissa ajureita, rajapintoja, käsittelijöitä ja johtajia. Ajureiden toimintaan kuuluu sisäisten ja ulkoisten laitteiden hallinta. Ajurit eivät kuitenkaan toimi sisällön välittäjinä, vaikka niillä käsitelläänkin tietoa. Rajapinnat on sijoitettu arkkitehtuurissa yleisesti ohjainlaitekerrokseen. Aju-

reiden ja käsittelijöiden tapaan rajapinnatkaan eivät osallistu tiedonvaihtoon. (Zurawski 2014: 1314.)

Käsittelijöiden tehtävänä on erilaisten asiakkaiden ajureihin tai rajapintamoduuleihin pääsyjen hallinnointi. Käsittelijä siis tarvittaessa puskuroi, jonottaa, sovittelee ja moninkertaistaa. Johtajat tarjoavat erityisiä palveluita useille asiakkaille. Niiltä tarvitaan, kun käsittelijöiden ollessa riittämättömiä rajapinta- ja ajurimoduulien käyttämiseen. Usein palvelukerrokselle sijoitettavat johtajat käsittelevät ja vaihtavat tai muokkaavat tietoa tarvittaessa. (Zurawski 2014: 1314.)

4 AUTOSARin viimeisimmät julkaisut ja vaiheet

4.1 Vaihe 1

AUTOSARin ensimmäisen vaiheen tavoitteena oli saavuttaa kokonaisvaltainen kokoelma arkkitehtuuria määritteleviä metodologioita ja malleja. Ensimmäiseen vaiheeseen lukeutuvat julkaisut 1–3.1, jotka ajoittuvat vuosien 2003 ja 2008 väliin. Ensimmäisen julkaisun konseptin perustavaiheen jälkeen julkaistu AUTOSARin julkaisu 1.0 määritelmistä käsitteli enimmäkseen osia perustavanlaisista ohjelmistoista käyttöympäristön tasolla. (Fürst: 2–3.)

Julkaisut 2.0 ja 2.1 erikoistuivat perusohjelmistojen komponenttien ja käyttöympäristön viimeistelyyn. Molemmat julkaisut ovat käytössä useiden AUTOSARin jäsenien toimesta sarjatuotannoissa. Julkaisu 2.1 on päivitys julkaisusta 2.0 ja se on valmis kokoelma määritelmiä ja se pitää sisällään kokoonpanokonseptin. Julkaisussa 2.1 uutena kokonaisuutena ovat täytöntöönpano ja validointi perusohjelmistojen moduulien sarjoille kuten laitteistojen alustoille. (Fürst: 4.)

4.2 Vaihe 2

AUTOSAR:in vaiheeseen 2 suunniteltiin kolmea julkaisua vuosina 2008 ja 2009. Tavoitteena oli jatkuva määritelmien kehittäminen ja uusien konseptien esitleminen. Julkaisu 3.0 julkaistiin alkuvuodesta 2008; ja se piti sisällään lukuisia parannuksia ja korjauksia edellisiin julkaisuihin pohjautuen. Julkaisu 3.1 puolestaan keskittyi yhteistyöhön OBD II -säädestä tukeviin mekanismeihin. Vaiheen 2 loppupuolella julkaistiin julkaisu 4.0, joka integroi uusia ominaisuuksia esimerkiksi turvallisuuteen ja viestintään liittyen sekä vaatimustenmukaisten testien määritelmät. (Fürst: 4.)

Laajahko 158-dokumenttinen julkaisu 3.0 sisältää noin lähes kolmasosan merkittäviä parannuksia ja 10 % dokumenteista on kokonaan uusia. Julkaisu sisältää yli puolen tuhatta muutosta, jotka parantavat standardin laatua. Perusohjelmistojen arkkitehtuuri on hyvin kattavasti määritelty. Perusohjelmistomoduulien kaupalliset täytöntöönpanot, jotka perustuvat julkaisuihin 3.0 ja 2.1 ovat markkinoilla. Tärkeimmät kehitysaskleet liittyvät ohjainlaitteiden ja verkkojen herättelyyn ja käynnistymiseen sekä käyttökohteiden monimutkaisuuden vähenemiseen. (Fürst: 4.)

Malleihin tehdyt parannukset mahdollistavat standardin johdonmukaisuuden. Rajapinnat, käyttäytyminen ja perusohjelmistojen kokoonpanojen parametrit ovat kyseisessä julkaisussa sisällytetty AUTOSARin malleihin noudattaen yhden lähteen periaatetta. Tällä mahdollistetaan hyvä ohjattavuus kehityksen myöhemmissä vaiheissa sekä automaattinen sukupolvi asianmukaisesti määritellyille kappaleille. AUTOSAR on myös työstänyt harmonisointia ASAM FIBEX -standardin ja AUTOSAR System Templaten välille. FIBEXin työkalut topologioiden kuvailuun, verkkoihin ja viestintään voidaan harmonisoinnin ansiosta integroida sujuvasti AUTOSARin metodologiaan ja työkaluihin. (Fürst: 5.)

Julkaisussa 3.0 on aloitettu käsittelemään telematiikan, multimedian ja ihmisen ja koneen välistä rajapinnan sekä matkustajien ja jalankulkijoiden turvajärjestelmiä. Näiden lisäksi voimalinja, alusta ja kori sekä mukavuus rajapintoja on integroitu alustavasti. Julkaisu sisältää sekä selittävät dokumentoinnit jokaisesta sovelluksesta että integroidun taulukon, jossa eritellään kaikki rajapinnat. (Fürst: 6.)

Julkaisu 3.1 oli suppeahko jatko julkaisu 3.0:lle. Siinä käsiteltiin uutena asiana OBD:n ominaisuuksien esittely AUTOSARin perusohjelmistomoduuleihin liittyen. Kaikki eri OBD-variantit sisältyivät tähän julkaisuun vaikuttaen lähinnä kolmeen perusohjelmistomoduuliin. (Fürst: 6.)

Vuonna 2009 valmistuneessa julkaisussa 4.0 esitellään uusia konsepteja metodologiaa ja arkkitehtuuriin. Esimerkiksi moduulitasolla suoritettavien vaatimuksenmukaisuus testien vaatimukset perusohjelmistoille on sisällytetty kyseiseen julkaisuun. Uusissa konsepteissa, jotka AUTOSAR julkaisu 4.0 esittelee on lisätty teknisiä ja käytännöllisiä parannuksia sekä laajennuksia eri alueisiin. Muun muassa toiminnalliseen turvallisuuteen, arkkitehtuuriin viestintään ja malleihin on kehitetty muutoksia. (Fürst: 6.)

AUTOSAR:in yhteen pääobjekteista, toiminnalliseen turvallisuuteen liittyen standardoinnissa on huomioitava ISO 26262 -standardi, kun käsitellään turvallisuuskriittisiä käyttökohteita. Esimerkiksi uutena tekniikkana muistin jaottelu-konsepti, joka tarjoaa vikasietoisen tekniikan turvallisuuskriittisten ja muiden sovelluksien erittelyyn toisistaan saman ohjainyksikön sisällä. (Fürst: 7.)

Mikrokontrollereille määritelty puolustava käyttäytymismalli puolestaan on luotu ratkaisuksi niihin kohteisiin, joissa ohjelmistollinen muistin jaottelun tuki ei ole käytettävissä.

Julkaisussa on myös tuki kaksoismikrokontrollereiden arkkitehtuurin vikojen havaitsemisen tavoitteisiin toissijaisen yksikön avulla. (Fürst: 7.)

Ohjelmistojen virran tarkkailulla on tarkoitus hallita sovellusten ajoitettua ja loogista käyttäytymistä tarkistamalla tiettyjä pisteitä koodin suoritteista tarvittaessa ajoituksellisten ja loogisten vaatimusten mukaisesti. Julkaisun 4.0 lopullisen kommunikoinnin suojaava kirjasto tarjoaa huipputekniikkaan turvallisuusprotokollan sovellustasolle. (Fürst: 7.)

Arkkitehtuurin kehitykset julkaisussa 4.0 näkyvät esimerkiksi moniydinkonseptissa, joka mahdollistaa AUTOSARin käsitellä mikrokontrollereita useammalla ytimellä ja tukea yhtenäisen sovelluksen muuttamista moniydinmikroprosessoriksi. Yksi lähestymistapa tulee olemaan itsenäinen käyttöjärjestelmän hallinta kaikille ytimille mikrokontrollerilla, jolloin uudet palvelut voidaan lisätä järjestyksessä osien aktivoimiseksi tai tietojen välittämiseksi ja eri objektien synkronointia varten. (Fürst: 7.)

Virheen käsittelykonsepti on kyseisessä julkaisussa harmonisoitu ja täydennetty uusiokäyttämään samaa strategiaa kuin muillakin arkkitehtuuriaalisilla alueilla, kuten muisti- ja viestintäosioissa. Kyseisessä tekniikassa mahdollistuu tiettyjen ohjelmistokomponenttien pysäyttely ja uudelleenkäynnistykset. (Fürst: 7.)

Julkaisun 4.0 mukanaan tuomat uudet konseptit ja aiemmat rajoitteet vaativat uutta lähestymistapaa ajoneuvon ja sovelluksien tilan hallinnan konsepteihin. Uusi tilan hallinta tarjoaa tilariippuvaisen hallinnan perusohjelmistojen resursseista. Tilan hallinnalla lisätään joustavuutta uusille laajennuksille mahdollistamalla, esimerkiksi: sovellusten ja ajoneuvon välisen tiedon jakamisella ja ristiriitaisen tilan selvittelyllä. (Fürst: 7–8.)

Viestintä osiota julkaisuun 4.0 on kehitetty ainakin lisäämällä AUTOSAR LIN -osioon tuki LIN 2.0:aan. Erityinen kokoonpanon parametri, jolla LIN 2.0:n toimintoja voidaan kytkeä päälle ja pois on esitelty julkaisussa. Julkaisuun 3.0 verraten julkaisuun 4.0 on tullut tuki suurille data tyypeille dynaamisine signaalinmitoin CAN- ja LIN-väylissä. Myös FlexRayhin liittyen sisältyy uutta kehitystä, kuten ajasta riippuvainen isännöinti väylällä ja joidenkin ominaisuuksien sisällyttäminen laitteistoon, minkä avulla tarjoutuu mahdollisuudet suorituskykyisille toteutuksille. (Fürst: 8.)

Julkaisussa 4.0 on kehitetty metodologeja ja malleja: lisäämällä mahdollisuus kuvailla ajoitusvaatimuksia, harmonisoimalla ohjainlaitteiden kokoonpanoparametrejä, parantamalla mittauksia ja kalibrointeja, ECU:n lähdemallien uusinnalla ja lähentymällä FI-BEX-standardin kanssa. Muunneltavuuden käsittely konseptilla turvataan mahdollisuus tukea eri muunnelmia eri tilanteissa. (Fürost: 8.)

Sovellusten rajapintojen standardointeja kaikilla ajoneuvon osa-alueilla sisältyy kattavasti AUTOSAR 4.0 julkaisuun. Erityisesti näiden standardin osien avulla mahdollistuu sovelluksien kierrättäminen, mistä seuraa ajoneuvoteollisuudelle säästöjä. Sovelluksien rajapintojen taulukko on eri AUTOSARin jäsenien yhteistyöllä kokoama ja sen avulla ohjelmistokehittäjät voivat uusiokäyttää tai laajentaa olemassa olevia ohjelmistokomponentteja erilaisia ECU:ja ja laitteistoja varten. (Fürost: 9.)

AUTOSAR ei standardisoi sovelluksien sisäistä käyttäytymistä ja toimintaa kuten algoritmeja ja optimointia, mutta sovelluksien välinen sisällönvälitys on määritelty. Tällä mahdollistetaan muun muassa eri tavarantoimittajien kehittämien sovellusten käyttäminen osana isompaa kokonaisuutta. (Fürost: 9.)

Julkaisun 4.0 validoinnit suoritettiin aiemmista julkaisuista poiketen jo ennen sen lopullista täytäntöönpanoa. Eri yritykset työstivät julkaisun vahvistamista pienissä osissa, jotta kokonainen lopullinen julkaisu olisi mahdollisimman laadukas ja toimiva. Tämä oli mahdollista, koska kyseinen julkaisu pohjautuu melko voimakkaasti aiempiin julkaisuihin, jolloin työkaluja ja osaamista oli saatavilla riittävästi. (Fürost: 9–10.)

4.3 Vaihe 3

Vuonna 2010 aloitettu vaihe 3 keskittyy olemassa olevien julkaisujen ylläpitoon ja toiminnallisuuksien valinnanvaran tehostamiseen. Vaiheen rakenteeseen kuuluu kehitystyön jakaminen teknisille ammattilaisryhmien hoidettaviksi projekteina. Vaiheeseen kuuluu julkaisun 5.0 kehittämistä samankaltaisesti, kuin julkaisu 4.0 kehitettiin validointien osalta. (Fürost: 13.)

Perusohjelmistojen arkkitehtuuriin ja moduuleihin on suunnitteilla vaiheessa 3 uusia toiminnallisuuksia kuten laitteiston lisätukia ja ristiinkäytettävyyttä AUTOSARin ja ajoneuvojen multimediasovellusten välillä. Olemassa olevia viestintämekanismeja on tar-

koitus kehittää erityisesti Ethernetin ja MOSTin osalta. Ohjainlaitteiden ja ohjelmistojen energiatehokkuuksiin kiinnitetään huomiota osittaisten verkkojen käsittelyllä. (Fürst: 13.)

Parannuksien tekeminen esimerkiksi turvallisuustekijöihin muistin jaottelulla ja diagnostiikkaan sekä kuorma-autojen erityispiirteiden huomioiminen ovat käsiteltävinä vaiheessa 3. Metodologiaan ja malleihin pyritään muun muassa tekemään lisäyksiä muunneltavuuteen, hajautettuun kehittämiseen, kokoonpanoon ja avoimeen tiedonvaihtoon. Suunnittelutyökalujen monikäyttöisyyden yksinkertaistaminen ja perusohjelmistojen uusien toiminnallisuuksien tukeminen ovat myös työstettävinä. (Fürst: 13.)

5 Väylätekniikka

5.1 Analogisesta tiedonsiirrosta digitaaliseen

Ajoneuvojen elektroniikan määrä on kasvanut eksponentiaalisesti 70-luvulta lähtien. Lukuisat elektroniset järjestelmät ovat syrjäyttäneet mekaanisia ja hydraulisia toteutuksia. Laitteistojen komponenttien kasvanut luotettavuus ja suorituskyky sekä niissä käytettävät ohjelmistot mahdollistavat lukuisia esimerkiksi mukavuuteen ja turvallisuuteen liittyviä käyttökohteita. (Zurawski 2014: 1269.)

Kun elektroniikka alkoi yleistyä ajoneuvoissa, kaikki uudet toiminnot toteutettiin itsenäisiä ohjainlaitteita hyödyntäen. Kyseiset osajärjestelmät koostuivat mikrokontrollerista, antureista ja toimilaitteista. Itsenäisten osajärjestelmien avulla toteutettavien toimintojen määrän ollessa rajallinen, ilmeni tarve kehittää järjestelmiä joissa eri ECUt kommunikoivat keskenään, jolloin uusia toimintoja saadaan aikaiseksi. Esimerkiksi moottorinohjainlaitteella arvioitu ajoneuvon nopeus tai pyyhkijän nopeuden automaattinen säätö. (Zurawski 2014: 1270.)

2010-luvun alun luksusajoneuvoissa käsitellään muutamia tuhansia signaaleja, joita ovat muun muassa perustiedot esimerkiksi ajonopeus. Näitä signaaleja välitetään jopa yli sadan ECU:n välillä. 1990-luvun alkuun saakka tiedonsiirto ohjainlaitteiden välillä tapahtui lävitse pisteestä pisteeseen linkkien lävitse. Tällainen kommunikointi ei kyennyt vastaamaan ohjainlaitteiden määrän kasvuun, koska ongelmaksi muodostuivat paino, kustannukset, luotettavuus ja monimutkaisuus. Ongelmia aiheuttivat muun muassa liittimien ja johtimien määrä. (Zurawski 2014: 1270.)

Edellä mainitut tekijät motivoivat ajoneuvoteollisuutta kehittämään verkkoihin perustuvaa tiedonsiirtoa ajoneuvoissa. Tästä puolestaan syntyi tarve määritellä säännöt väylätiedonsiirrolle, jotta tiedonsiirtoa voidaan hallita ja väylän tietoihin ylipäänsä saadaan yhteys. Bosch kehitti 80-luvun lopulla CAN-väylätekniikan, joka otettiin 90-luvun alussa käyttöön ensimmäisen kerran ajoneuvotuotannossa Mercedeksen toimesta. CAN-väylätekniikka on vielä nykyään yleisin väyläratkaisu, vaikka sen rinnalle on tullut muitakin tiedonsiirtoverkkoja. (Zurawski 2014: 1270.)

Esimerkkinä saavutetusta painonsäästöstä perinteisen johdotuksen korvaamisesta lähiverkkotekniikalla, BMW kevensi vuonna 1998 neljän oven kokonaismassaa 15 kg:n verran (Zurawski 2014: 1270). 2010-luvun alussa ECUja voi olla premium-autoissa lähemmäs sata ja niiden yhdistämisiin käytetään viittä eri väyläratkaisua, jotka mahdollistavat yli 800 erilaista toimintoa (Zurawski 2014: 1299.)

5.2 Eri väylät vaatimuksineen

Kaistanleveyden kasvun tarve on ollut tasaisesti kiihtyvää läpi väylätekniikan historian. Esimerkiksi vuonna 1994 kaistanleveyttä vaadittiin maltilliset 122 kbit/s alustan ja moottorin hallintaan. Kymmenessä vuodessa kaistanleveys kasvoi arvoon 765 kbit/s ja siitä neljä vuotta eteenpäin vaadittavan kaistanleveyden arvioitiin lähes kaksinkertaistuvan arvoon 1500 kbit/s. Kaistanleveyden kasvun, suorituskyvyn monipuolistumisen, kustannusten ja osien saatavuuden vaatimuksien johdosta ajoneuvojen verkottuminen on monipuolistunut merkittävästi. (Zurawski 2014: 1272.)

Ajoneuvojen verkkojen kommunikointiprotokollat luokiteltiin vuonna 1994 SAE:n (Society for Automotive Engineers) toimesta. Luokittelu perustuu tiedon siirron nopeuteen ja toimintoihin joita välitetään verkon välityksellä. Luokkia on kolme kappaletta, luokat A, B ja C, joista A on hitain ja C nopein. Lisäksi on epävirallinen luokka D, johon usein kuitenkin viitataan. (Zurawski 2014: 1272.)

Luokan A verkot toimivat nopeudella alle 10 kbit/s ja ovat verrattain edullisia toteuttaa. Luokan A verkkoja ovat esimerkiksi LIN ja TTP/A, joita käytetään muun muassa valaistuksessa, ovien lukituksessa ja istuimen säätimissä. (Zurawski 2014: 1272.)

Luokan B verkot tukevat tiedon vaihtoa ECU:jen välillä tavoitteena säästää moninkertaista anturointia vaihtamalla anturitietoja keskenään. Esimerkiksi low-speed CAN ja J1850 ovat tämän luokan verkkoja, joiden kaistanleveys vaihtelee välillä 10 ja 125 kbit/s. (Zurawski 2014: 1272.)

Korkeita nopeuksia reaaliaikaiseen viestintään tarvitsevat sovellukset vaativat luokan C- tai D-verkkoja. Luokan C tiedonsiirtonopeus on enemmän kuin 125 kbit/s mutta vähemmän kuin 1 Mbit/s. Luokan D tiedonsiirtonopeudet puolestaan ovat vähintään 1 Mbit/s. (Zurawski 2014: 1273.)

Luokan C verkot kuten high-speed CAN ovat usein käytössä voimalinjan tai alustan sovelluksissa. Luokan D verkkoja kuten MOST ja FlexRay käytetään lähinnä multimedial siirrossa. Nykyaikaisille ajoneuvoille on tyypillistä, että niissä on käytetty useammanlaista verkkoa ja samanlaisia verkkojakin voi olla useampi rinnakkain. (Zurawski 2014: 1273.)

5.3 LIN

AUTOSAR-standardi kattaa LIN-protokollan isäntäsolmun osalta. LIN-orjasolmut ovat rajattu standardin ulkopuolelle. Väyläteknisesti AUTOSAR LIN protokolla on samanlainen kuin LIN-määritelmä. Ohjelmistovaatimukset alkuperäisen LIN-määritelmän ja AUTOSAR LINin välillä ovat erilaiset, johtuen AUTOSARin omanlaisesta arkkitehtuurista. Toisin kuin täytäntöönpano APLa, AUTOSARissa tuetaan täytäntöönpanossa aikataulutaulukoita. (Zurawski 2014: 1403.)

AUTOSAR LIN-määritelmiä ovat LIN-liitäntäohjelmistojen määritelmä, LIN-verkon hallinta, LIN-tilan hallinta, LIN-ajuri ja LIN-lähetinjuri. LIN-liitäntäohjelmistojen määritelmä kattaa määritelmät isäntäsolmun käyttäytymiseen ja kytkeytymisiin AUTOSAR-arkkitehtuuriin. LIN-tilan hallinta käsittelee tilan vaihtomekanismeja kyseisellä väylällä. LIN-ajuri puolestaan liittyy mikrokontrollereiden kuvailuun ja LIN-lähetinjuri käsittelee monimutkaisia LIN-lähettäjiä. (Zurawski 2014: 1403.)

6 Viestintä ja tiedonsiirto

6.1 Partial ja pretended networking

Energiatehokkuus on tärkeä huomioitava tekijä suunniteltaessa sähköjärjestelmiä ajoneuvoihin. Keskimääräinen hiilidioksidipäästö määrä EU:ssa tulisi olla alle 95 g/km vuodesta 2020 lähtien. Vuosien 2002 ja 2012 välillä ajoneuvoelektroniikan virrankulutus on nelinkertaistunut kolmeen kilowattiin. (Zurawski 2014: 1273.)

Päästrategioina energiatehokkuuteen ovat partial networking (osittainen verkostoituminen) ja ECU:n alennus. ECU:n alentamisella sammutetaan käynnistys portteja, vähennetään ECU:n suorituksia tai jopa kytketään ohjainlaite lepotilaan jos mahdollista. AUTOSAR tukee jo näitä matalan virrankulutuksen ominaisuuksia. (Zurawski 2014: 1273.)

Partial networking tarkoittaa sitä, että väliaikaisesti laitetaan lepotilaan kokonaisia solmuja tai koko verkosto, kun niiden ei tarvitse olla käytettävissä. Esimerkiksi pysäköintiavustin, jota ei voi käyttää yli 20 km/h nopeuksilla voidaan kytkeä pois korkeammilla ajonopeuksilla. Useat korin toiminnot kuten istuimet, peilit ja lukot voivat olla suurimman osan ajasta lepotilassa, ja kun niitä tarvitaan, voidaan solmut herättää. Audin arvioiden mukaan kyseisellä tekniikalla voidaan säästää hiilidioksidipäästöissä noin 2,6 g/km. (Zurawski 2014: 1273.)

Partial networking on ollut kehitettävänä vuodesta 2008, ja se on jo täysin tuettu AUTOSAR for CANissa. Useita komponentteja jotka tukevat osittaista verkostoitumista on tuotannossa ja vuonna 2014 tuotettiin ajoneuvoja joissa tekniikkaa hyödynnetään. (Zurawski 2014: 1274.)

Partial- ja pretended networking energian säästömekanismit ovat eritelty AUTOSAR Efficient Energy Management Groupin toimesta. Erona alkuperäiseen CAN-standardiin partial networkingissä ohjainlaitteen solmuja voidaan kytkeä alemman virrankulutuksen tilan sijaan syvempään lepotilaan. Myös vain tiettyjen solmujen käynnistäminen on mahdollista toisin kuin alkuperäisessä CAN-standardissa, jossa koko väylä herätettiin. (Hong ym.: 2.)

Pretended networking puolestaan tarkoittaa sitä, että mikrokontrollerin toiminta pysäytetään, jolloin energiankulutus alenee. Mekanismin etuna on se, että sitä voidaan käyttää turvallisuuskriittisemmissä ja sujuvaa toimintaa vaativissa kohteissa, koska pysäytetty mikrokontrolleri käynnistyy huomattavasti nopeammin kuin virraton. Mekanismin hintana on kuitenkin partial networkingiä selvästi korkeampi virrankulutus, jonka vuoksi se ei ole paras ratkaisu kaikissa sovelluksissa. (Hong ym.: 2.)

6.2 Ethernet

Ethernetin hyödyntäminen ajoneuvoissa tarjoaa edullista ja suhteellisen valmista teknologiaa. Kaistanleveyttä Ethernetin välityksellä on mahdollista hyödyntää enemmän kuin kilpailevilla menetelmillä. Tämä on tärkeätä aktiivisen turvallisuuden sovelluksissa ja tietojärjestelmissä. Ajoneuvotekniikan ulkopuolella käytössä olevia sovelluksia voidaan helpohkosti uusiokäyttää, kunhan ajoneuvotekniset rajoitukset huomioidaan. Esimerkiksi partial networking energian säästämiseksi, virransyöttö Ethernetin yli johdotuksen minimoimiseksi sekä käyttöympäristön vaatimuksien huomioiminen. (Zurawski 2014: 1286.)

Ensimmäisen sukupolven Ethernet-verkko, joka on ollut käytössä vuodesta 2008, on tarkoitettu diagnostiikkaa ja koodin lataamista varten. Toisen sukupolven Ethernet verkko vuodesta 2015 eteenpäin tukee tietojärjestelmiä ja kameraan pohjautuvaa kuljettajaa avustavaa järjestelmää. AUTOSAR-version 4.1 myötä Ethernet-tuki julkaistiin ajoneuvon sisäiseen viestintään liittyen (Zurawski 2014: 1309.) Kolmannen sukupolven Ethernetin on arvioitu tulevan vuonna 2020. Siinä nopeus on gigabitin verran, ja sillä voinee korvata nykyisenlaisia yhdyskäytäviä ja yhdistää muita verkkoja keskenään. (Zurawski 2014: 1286.)

6.3 Domain-pohjaiset verkkorakenteet

Vastauksena hajautettujen väyläjärjestelmien ongelmiin ovat domain-pohjaiset verkkorakenteet. Hajautetuissa väyläjärjestelmissä pienitehoisia ohjainlaitteita on moninkertainen määrä verrattuna domain- eli toimialuepohjaisessa järjestelmään, jossa suorituskkyisiä ohjainlaitteita on lukumääräisesti vähemmän. Domain-pohjaisissa verkoissa

toimintoja hajautetaan toimialakohtaisesti. Toimialueita ovat esimerkiksi ajodynamiikka, turvajärjestelmät, mukavuuselektroniikka ja kori. (Pasanen 2016: 62.)

Ohjainlaitteiden määrän vähenemisellä edesautetaan tiedonsiirron nopeutumista, järjestelmän toimintavarmuutta ja rakenteen yksinkertaistumista. Tiedonsiirtoviiveiden lyhentyminen on hyödyllistä erityisesti turvallisuuskriittisissä toiminnoissa. Domain-pohjaisissa järjestelmissä hyödynnetään runkoverkkoina kaistanleveysvaatimuksien vuoksi lähinnä Ethernetiä tai Flexraytä. Eri toimialueita ohjailevat ohjainyksiköt on liitetty järjestelmään runkoverkon avulla. (Pasanen 2016: 62.)

Domain-pohjaisien verkkojen virrankulutusta optimoidaan hyödyntämällä virransäästötiloja mahdollisuuksien mukaan. Tällä pienennetään päästöjä ja polttoaineenkulutuksessa saavutetaan parhaimmillaan säästöä jopa 0,1–0,2 l / 100 km. (Pasanen 2016: 62.)

Virransäästöä toteutetaan kytkemällä väyliä virransäästötiloihin. Turvallisuuskriittisissä toiminnoissa, joiden on oltava toiminnassa jatkuvasti ajon aikana, ei virransäästöä voida käyttää. Kyseiset toiminnot ovat integroitu toimialaa hallitsevan ohjainlaitteeseen, jolloin vaarallisilta viiveiltä ja toimimattomuuksilta vältytään. (Reinhardt & Kucera: 223.)

Sähkölaitteiden virransäästö ja yhä monimutkaistuvat toiminnot ajoneuvoissa aikaansaavat tarpeen diagnostiikkalaitteille. Ilman näitä diagnostiikka olisi mahdotonta, sillä perinteisillä mittauksilla ei selviä välttämättä onko jokin komponentti rikki vai virransäästötilassa. (Pasanen 2016: 62.)

Yhtenä merkittävänä etuna domain-pohjaisissa verkoissa on johdotusten merkittävä väheneminen ohjainlaitteiden määrän ollessa maltillinen. Johdotusten minimoinnilla saavutetaan kustannussäästöjä esimerkiksi materiaalin kulutuksen, massan vähenemisen ja asennusaikojen lyhentymisen seurauksena. Johdotusten optimointia rajoittaa muun muassa se, että useimpiin toimilaitteisiin ja antureihin tarvitaan kuitenkin fyysinen yhteys. (Reinhardt & Kucera: 222.)

Merkittävä haittavaikutus domain-pohjaisella verkkorakenteella on se, että vaatimukset käytettävien ohjainlaitteiden suorituskyvylle nostavat niiden hintoja. Järjestelmään liittyvät korjaustoimenpiteet ovat tästä johtuen kalliimpia, kuin hajautetussa järjestelmässä, jossa käytetyt tarvittaessa vaihdettavat komponentit olivat edullisempia. Haittapuolena

on myös se, että yhden ohjainlaitteen vioittuessa suurempi osa mahdollisesti turvallisuuskriittistä järjestelmää lopettaa toimintansa. (Reinhardt & Kucera: 222–223.)

6.4 Viestintäpalvelut

AUTOSARin standardointien yksi päätavoitteista on laitteiston sisäisen rakenteen piilottaminen ja sovellusohjelmien jakelu kustannustehokkaasti. Tämän vuoksi ohjainlaitteiden väliset ja/tai sisäiset standardoidut viestintäpalvelut ovat tärkeitä ajoituksellisuuden ja käyttäytymisen hallinnallisesti. Näiden palveluiden suunnittelu, tuottaminen ja kokoonpaneminen tulee olla tarkasti hallittua ja etenkin ajoitukselliset tekijät tulee varmistaa tarkasti määrätynlaisiksi. (Zurawski 2014: 1290.)

Viestittävät asiat kuten signaalit ja viestikehykset, joita käsitellään eivät ole samanlaisia eri tasoilla. Tästä huolimatta jokainen käsiteltävä kohde on voimakkaasti riippuvainen useammistakin naapuritasoilla käsiteltävistä kohteista. AUTOSAR-standardissa esitellään kaksi erilaista viestinnän mallia: lähettäjä-vastaanottajamalli, jota käytetään tiedonvälitykseen kahden sovellusohjelmakomponentin välillä, ja asiakas-palvelinmalli, joka tukee toiminnon pyyntöä. (Zurawski 2014: 1290.)

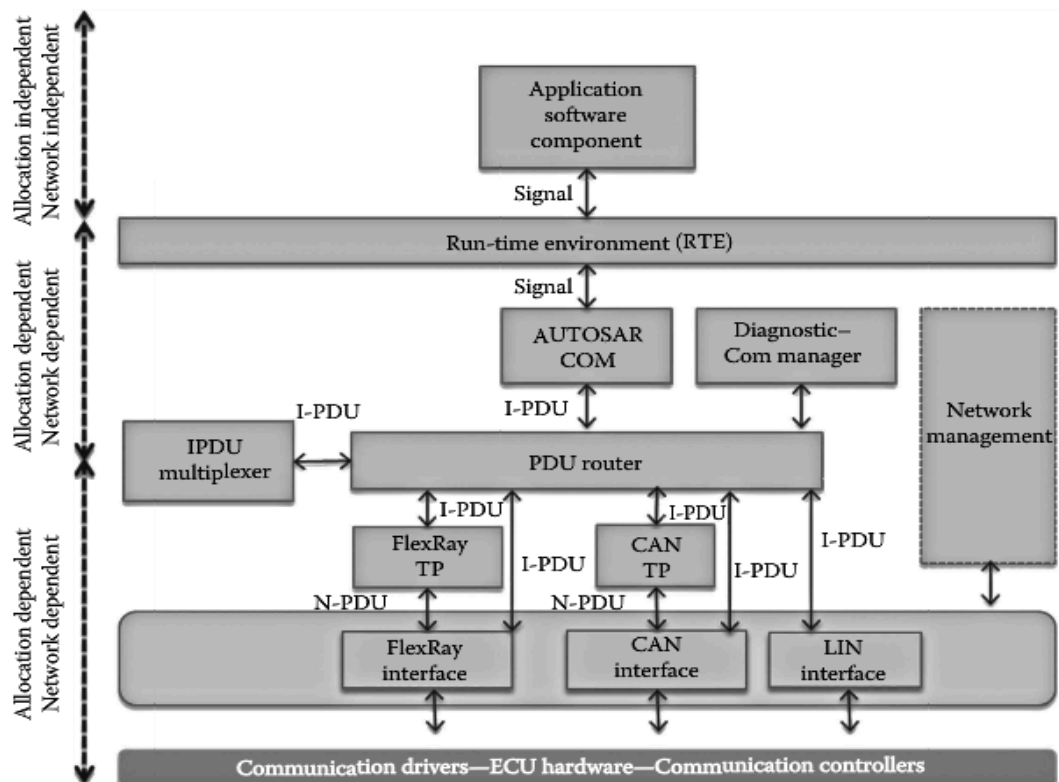
Lähettäjä-vastaanottajamallissa on kaksi erilaista viestintätilaa tuettuina: epäsuora tila ja täsmällinen tila. Täsmällistä tilaa käytetään komponentille, joka tekee täsmällisiä kutsuja tiedon lähettämiseksi ja vastaanottamiseksi. Epäsuora tila puolestaan tarkoittaa sitä, että tiedon luenta suoritetaan automaattisesti ennen pyynnön lähettämistä tietoa kuluttavalta komponentilta ilman tarkkaa kutsua AUTOSARin palveluille. Tällä tavoin suojataan tehokkaasti sovellusohjelmalta eteenpäin lähtevä tieto. (Zurawski 2014: 1291.)

AUTOSAR erittelee kolme pääkohdetta viestinnässä: ohjelmistokomponenttien välillä siirrettävät signaalit, I-PDU, jossa käsitellään useita signaaleita, ja N-PDU, joka lähettää viestit väylälle. Sovelluksien tasolla AUTOSAR määrittelee signaalin pituudet ja tyypit. Viesti välitetään sovellusohjelmien välillä porttien lävitse välittämättä komponentin jako-ominaisuuksista. Sovelluksen täytyy ilmaista siirron ominaisuusparametrit, jotka vaikuttavat siirron käyttäytymiseen ja joiden arvoja välitetään eteenpäin tai jätetään vireille. Viestiä suunniteltaessa on myös määritettävä, onko se tietoa, tapahtuma vai

tila, jotta sen käsittely saadaan priorisoitua sopivasti ruuhkautumisen välttämiseksi. (Zurawski 2014: 1291.)

AUTOSARin RTE käsittelee erityisesti lähettäjä-vastaanottajamallissa täsmällisiä ja epäsuoria viestintätiloja ja säätelee viestien sisältöjen jonouttamista tai suoraa käsitte-lyä sisällöstä riippuen. AUTOSAR COM -komponentti puolestaan on vastuussa esi-merkiksi lähettäjälle ja vastaanottajalle ilmoitettavista viestin onnistuneesta tai epäon-nistuneesta tiedon välityksestä määrätyn ajan sisällä. COM-komponentin tehtävänä on myös viestien suodattaminen I-PDU:lta sovellukselle tai portille ja viestien pakkaami-nen ja purkaminen I-PDUiden tiedonsiirrossa. (Zurawski 2014: 1292.)

COM-komponentit on tuotettu erillään, mutta siihen on sisäänrakennettu tietoisuus viesteistä, I-PDUista ja ohjainlaitteiden sovellusohjelmista jotka ovat tietoa jakamassa. AUTOSAR PDU -reitittimen tehtävänä on jakaa jokainen I-PDU oikeaan verkon viestintäosioon. Muut perusohjelmistokomponentit viestinnän osiossa ovat vastuussa I-PDUiden osiin jakamisesta väylistä riippuen tarvittaessa tai rajapintojen toimittamises-sa viestintäajureille eri väyläajureiden tapauksissa. Kuvassa 3 on esitetty AUTOSARin viestintäarkkitehtuuri (Zurawski 2014: 1291.)



Kuva 3. Kuvaus AUTOSARin viestinnän arkkitehtuurista. (Zurawski 2014: 1289).

6.5 Tietoturvallisuus

Ajoneuvojen sähköjärjestelmien turvallisuuden merkitys kasvaa sitä mukaan, kun ajoneuvoja liitetään internetiin, toisiinsa, infrastruktuuriin tai telematiikkaan. Olemassa olevat turvamekanismit ovat nykyisellään kovin heikkoja. Esimerkiksi ohjainlaitteiden uudelleenohjelmoinnin kontrollointi on vähäistä ja ajoneuvon järjestelmät eivät usein vastusta hyökkäyksiä tarpeeksi. Esimerkiksi OBD-pistokkeen kautta on mahdollista ohjelmoida ohjainlaitteita turvallisuuskriittisten tekijöiden kuten moottorin ja jarrujen osaltakin. (Zurawski 2014: 1274.)

Ohjainlaitteiden uudelleen ohjelmoitavuus on tärkeätä ajoneuvovalmistajille ohjelmistojen päivittämisen kannalta. Ohjainlaitteiden ohjelmointikoodeja voidaan päivittää perinteisesti OBD-pistokkeen kautta laiteohjelmistojen kehittämiseksi. Päivitysmenetelmien kehittyessä langattomampaan suuntaan tietoturvallisuuden merkitys korostuu. Langallisesti päivitettävien ajoneuvojen merkittävimpiä tietoturvariskejä ovat lähinnä ohjainlaitteiden lastutuksella aikaansaavat toiminnan muutokset. (Weimerskirch 2009: 1.)

Ohjainlaitteiden muokkaamista kolmansien osapuolien ohjelmilla pyritään estämään digitaalisilla allekirjoituksilla, jotka integroidaan ohjainlaitteisiin. Virallisiin päivityksiin ajoneuvovalmistajat liittävät vastaavat allekirjoitukset, jolloin päivitykset voidaan vahvistaa ja ladata ohjainlaitteisiin. Näistä on hyötyä, mikäli laitteistoon luvattomasti pääseminen on estetty esimerkiksi kokoonpanollisesti, eikä avoimia takaovia vahvistamattomille päivityksille ole jätetty. (Weimerskirch 2009: 1.)

Asiakaspalvelun parantamiseksi ajoneuvoissa yleistyvät langattomat vianhaku ja viankorjaus mahdollisuudet. Ajoneuvojen langattomaan diagnostiikkaan liittyy kuitenkin uhkia kyberhyökkäyksistä kohdistuen esimerkiksi ajoneuvon hallintaan liittyen yhä itenäisemmin liikkuvissa ajoneuvoissa. Myös yksityisyydensuojan kannalta merkittäviä tietoja jotka voivat olla pilvipalveluissa, tulee suojella kolmansilta osapuolilta. (Salfer & Eckert 2015: 1.)

Teknisenä vastauksena ajoneuvojen sähköjärjestelmien suojaamiseksi esimerkiksi AUTOSAR 4.0 -julkaisussa on tuki salausmoduulille, jota hyödyntämällä turvallisuustasoa voidaan nostaa. Kaistanleveysvaatimuksista johtuen CAN FD (CAN with Flexible Data-Rate) toimii tavallista CAN-väylää sujuvammin kyseisessä tapauksessa. (Zurawski 2014: 1273.)

AUTOSAR 4.2.1 -julkaisussa esitellään turvallisen sisäisen viestinnän moduuli, jonka tarkoitus on lisätä varmennusmekanismeja standardointeihin. Kryptografisia menetelmiä hyödyntämällä pyritään torjumaan uusia uhkia tietoturvallisuudessa. Julkaisussa on standardoitu kahden ohjainlaitteen välinen viestiliikenne siten, etteivät kolmannet osapuolet pääse pysäyttämään tai vaikuttamaan siihen. (Basic Software for AUTOSAR 2015: 1.)

7 Kerätyn aineiston analysointi

Työssä kerätty aineisto tarjoaa opetustarkoituksiin hyödynnettävissä olevaa materiaalia AUTOSAR-arkkitehtuurista. Tutkimuskohteesta löytyi tietoa tavoitteiden mukaisesti. Väylätekniikasta tietoa oli runsaasti saatavilla, joten sitä tiivistettiin melko reippaasti keskittyen AUTOSARin kannalta merkittäviin asioihin. AUTOSAR-arkkitehtuurista materiaalia oli saatavilla useista eri lähteistä; tosin materiaalit olivat sisällöiltään melko samankaltaisia keskenään esimerkiksi arkkitehtuurikerrosmallin osalta.

AUTOSARin standardeista materiaalia oli saatavilla kohtalaisesti, vaikka julkisten lähteiden käyttö rajoitti standardijulkaisujen tutkimisen informatiivisiin tarkoituksiin rajoittuneisiin versioihin ja niistä kertoviin dokumentteihin. Tietoa standardeista saatiin kerättyä aiheeseen tutustumiseen yleisellä tasolla riittävän kattavasti. AUTOSAR-organisaatiosta kerätty materiaali vastaa hyvin tavoiteltua laajuutta, ja sen avulla on muodostettavissa yleiskuva organisaation tärkeimmistä vaiheista ja toiminnasta.

Partial ja pretended networkingistä sekä domain-pohjaisista verkkorakenteista työhön kertyi aineistoa lähinnä aiheiden esittelyn ja peruspiirteiden selvittämisen muodossa. Näiden vianhakumenetelmiinkin vaikuttavien tekniikoiden opetus esimerkiksi autosähkötekniikan opiskelijoille diagnoositekniikassa voisi olla hyödyllistä. Aiheeseen liittyvässä opiskelussa olisi mahdollisesti havainnollistavaa suorittaa nykyaikaista välineistöä vaativia laboratorioharjoitteitakin.

Ajoneuvoteollisuuden ohjelmistokehittäjille AUTOSARin syvällisempikin tuntemus muun ohjelmointiosaamisen lisäksi on varmasti tärkeätä. Suomen työmarkkinoilla ohjainlaitteiden ohjelmointiin liittyviä työtehtäviä on rajallisesti tarjolla, joten aiheen kattavampi opetus tuskin on tarpeellista.

8 Yhteenveto

Työn tuloksena syntyi tavoitteena ollut suomenkielinen materiaalipaketti AUTOSARista ja siihen liittyvistä ajankohtaisista tekniikoista. Materiaalia on mahdollista käyttää aiheen opiskelussa. Materiaalin avulla aiheeseen tutustuminen on nopeampaa, kuin eri lähteitä etsien ja tutkien.

Työn tuloksena selvisi, että AUTOSAR on ajoneuvojen sähköjärjestelmiä kehittävien yritysten muodostama organisaatio, joka kehittää standardoitua AUTOSAR-arkkitehtuuria. Standardoinnin kehittämisen motiivina on suunnittelu- ja tuotantokustannuksien minimointi järjestelmien komponenttien käyttömahdollisuuksia kasvattamalla. Arkkitehtuuria on kehitetty kolmessa vaiheessa, joissa on julkaistu päivitettyjä versioita standardista.

AUTOSARin standardeissa on huomioitu yhteensopivuudet eri väylien ja niissä käytettävien uusienkin tekniikoiden välillä. AUTOSARin standardeja kehitetään jatkuvasti, sitä mukaan, kun uusien toimintojen myötä kehitysvaatimuksia ilmenee. Tämän vuoksi myös aiheen tutkimista kannattaa tulevaisuudessa jatkaa erityisesti päivittyneiden ja uusien tekniikoiden osalta, jotta tietämys niistä pysyy ajan tasalla.

AUTOSARin syvällinen tunteminen lienee merkittävässä roolissa, mikäli työskentelee ajoneuvoteollisuudessa ohjelmistokehityksen parissa. Työssä käsiteltyjen uudehkoja tekniikoiden kuten partial- ja pretended networking sekä domain-pohjaiset verkkorakenteet tuntemisesta voi olla hyötyä myös ajoneuvojen diagnostiikan kanssa työskentelevillekin. Muutoin jälkimarkkinoinnin parissa työskenteleville AUTOSAR-arkkitehtuurin syvällisellä tuntemuksella ei vaikuta olevan tarvetta.

Lähteet

Basic Software for AUTOSAR 4.2.1 Already Available. 2015. Verkkoaineisto Vector Informatik GmbH.

<https://vector.com/vi_autosar_news_detail_en,1371902,,1355155,detail.html>. Luettu 1.2.2018.

Bosch Automotive Electrics and Automotive Electronics. 2007. Robert Bosch GmbH Ed. Viides painos. Plochingen. Robert Bosch GmbH.

Current Partners. 2017. Verkkoaineisto. AUTOSAR.

<<https://www.autosar.org/about/current-partners/>>. Luettu 5.11.2017.

Eckert, Claudia & Salfer Martin. 2015. Attack surface and vulnerability assessment of automotive Electronic Control Units. Verkkoaineisto. <<Http://ieeexplore.ieee.org.ezproxy.metropolia.fi/document/7518052/>>. Luettu 7.2.2018.

Fürst Simon. AUTOSAR – A Worldwide Standard is on the Road. Verkkoaineisto. <<http://www.win.tue.nl/~mvdbrand/courses/sse/0809/papers/AUTOSAR.pdf>>. Luettu 8.1.2018.

Hong, Wei; Lin Juguang & Viehl, Alexander. 2016. Impact analysis of AUTOSAR energy saving mechanism for automotive networks. Verkkoaineisto. <<http://ieeexplore.ieee.org.ezproxy.metropolia.fi/document/7535526/>>. Luettu 31.1.2018.

Kovanen Pasi. 2017. Lehtori, Metropolia Ammattikorkeakoulu Oy, Helsinki. Keskustelu 12.10.2017.

Kucera, Marcus & Reinhardt, Dominik. Domain Controlled Architecture. Verkkoaineisto. <<https://pdfs.semanticscholar.org/65ff/f1cd276736bc5cf67d0cb30db269cd08b5f5.pdf>>. Luettu 31.1.2018.

Learning Module AUTOSAR. 2016. Verkkoaineisto. Vector Informatik GmbH. <https://elearning.vector.com/index.php?seite=vl_autosar_introduction_en&root=378422&wbt_ls_kapitel_id=1330159&wbt_ls_seite_id=1330160&d=yes>. Luettu 30.10.2017.

Maradana, Sudhakar. AUTOSAR BASICS. 2015. Verkkoaineisto. Automotive techies. <<https://automotivetechis.wordpress.com/autosar-concepts/>>. Luettu 1.2.2018.

Pasanen, Lauri. 2016. Väylätekniikka murroksessa. Suomen autolehti 6/2016.

Standards. 2017. Verkkoaineisto. AUTOSAR. <<https://www.autosar.org/standards/>>. Luettu 6.11.2017.

Weimerskirch, André. 2009. Secure Software Flashing. Verkkoaineisto. <<https://www.sae.org/publications/technical-papers/content/2009-01-0272/>>. Luettu 6.2.2018.

Zurawski, Richard. 2014. Industrial Communication Technology Handbook, 2nd Edition. E-Kirja. CRC Press.

